Introduction

Our project is all about understanding how different factors contribute to an individual's diagnosis of diabetes. We want to identify the primary factors in preventing and managing diabetes. To accomplish this, we'll analyze the data to investigate the impacts of socioeconomic factors, such as education and income, as well as lifestyle choices like dietary habits and physical activity levels, on the prediction of diabetes. With this project, we hope to discover important associations that can help individuals to make informed choices influencing their likelihood of being diagnosed with diabetes.

Questions of Interest and Contributions

What socioeconomic factors and lifestyle choices have a significant impact on an individual's diabetic status? - Menaka Gc and Theodore Lam

What do the intra-relationships look like within these socioeconomic factors and lifestyle choices? - Shriveda Reddy

Given the problem of multicollinearity in predicting models, should some factors be excluded as predictors to increase the efficiency of the model? - Justin Lee

Dataset

Our dataset from Kaggle contains healthcare statistics and the answers pertaining to the lifestyles of 70,692 individual, including their diabetes diagnosis status. Derived from the 2015 Behavioral Risk Factor Surveillance System (BRFSS), a telephone survey conducted by the CDC, the dataset is named "diabetes_binary_5050split_health_indicators." This dataset is balanced, featuring an equal number of individuals without diabetes and those diagnosed with either diabetes or prediabetes.

Methodology

To analyze our questions of interest, we will be utilizing logistic regression to analyze relationships between predictors and response variables and creating models. To test these models, we will be using k-fold cross validation to see which model produces the lowest cross validation error.

What lifestyle choices have a significant impact on an individual's diagnosis of diabetes?

```
##
## Call:
## glm(formula = Diabetes_binary ~ PhysActivity + Fruits + Veggies,
##     family = binomial, data = data)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.73095    0.02037  35.878  < 2e-16 ***
## PhysActivity -0.66003    0.01710 -38.588  < 2e-16 ***
## Fruits       -0.09221    0.01620  -5.693 1.25e-08 ***
```

```
## Veggies      -0.26328    0.01948 -13.514  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 98000  on 70691  degrees of freedom
## Residual deviance: 95944  on 70688  degrees of freedom
## AIC: 95952
##
## Number of Fisher Scoring iterations: 4
```

This is the summary table from our logistic regression with diabetes_binary being our response variable and the variables physactivity, fruits, and veggies being our predictors. As seen from the summary table, the p-value for all the predictors are smaller than any common alpha (0.01,0.05,0.1), hence we can conclude that all three predictors are statistically significant.

```
##
##  Welch Two Sample t-test
##
## data:  PhysActivity by Diabetes_binary
## t = 42.727, df = 69241, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  0.1383438 0.1516466
## sample estimates:
## mean in group 0 mean in group 1
##       0.7755333       0.6305381
```

The t-test summary table above is the Welch two sample t-test conducted with the response variable being diabetes_binary and the predictor variable being an individual's physical activity level. The responses to the predictor being 0 (no) or 1 (yes) for having done physical activity in the past month. Mean (of physical level activity) in Group 0 (no diabetes): 0.7755333. Mean (of phsical level activity) in Group 1 (diabetes or prediabetes): 0.6305381 The mean in group 0 is higher than the mean in group 1. Therefore, in the context of physical activity (where a higher value indicating more physical activity), individuals without diabetes (group 0) have, on average, higher levels of physical activity compared to individuals with diabetes (group 1).

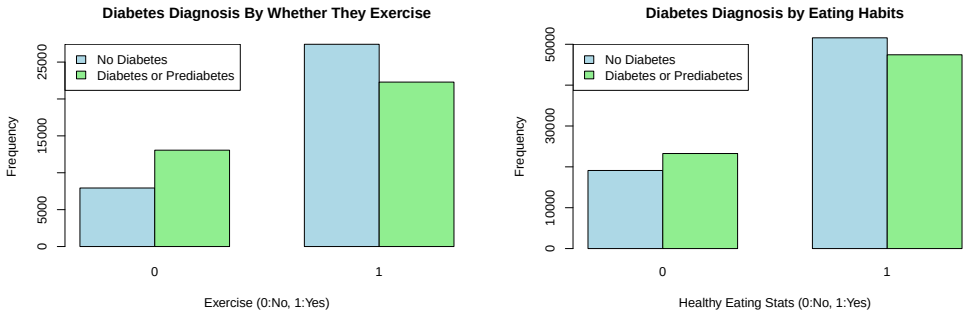
```
##
##  Welch Two Sample t-test
##
## data:  Veggies by Diabetes_binary
## t = 21.149, df = 69800, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  0.05873240 0.07073063
## sample estimates:
## mean in group 0 mean in group 1
##       0.8211396       0.7564081
```

The t-test summary table above is the Welch two sample t-test conducted with the response variable being diabetes_binary and the predictor variable being an individual's veggie intake. The responses to the predictor being 0 (no) or 1 (yes) for having eaten vegetable at least once a day. Mean (of whether individuals eat

veggies) in group 0 (no diabetes): 0.8211396. Mean (of whether individuals eat veggies) in Group 1 (diabetes or prediabetes): 0.7564081. The mean in group 0 is higher than the mean in Group 1. Therefore, in the context of veggies intake (where a higher value indicates more veggie intake), individuals without diabetes (group 0) have, on average, higher levels of veggie intake compared to individuals with diabetes (group 1).

```
##
##  Welch Two Sample t-test
##
## data:  Fruits by Diabetes_binary
## t = 14.399, df = 70646, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  0.04553281 0.05988223
## sample estimates:
## mean in group 0 mean in group 1
##       0.6381486       0.5854411
```

The t-test summary table above is the Welch two sample t-test conducted with the response variable being diabetes_binary and the predictor variable being an individual's fruit intake. The responses to the predictor being 0 (no) or 1 (yes) for having eaten fruit at least once a day. Mean (of whether individuals eat fruits) in group 0 (no diabetes): 0.6381486. Mean (of whether individuals eat fruits) in Group 1 (diabetes or prediabetes): 0.5854411. The mean in group 0 is higher than the mean in Group 1. Therefore, in the context of fruit intake (where a higher value indicates more fruit intake), individuals without diabetes (group 0) have, on average, higher levels of fruit intake compared to individuals with diabetes (group 1).



From this left barplot, we can see that those who exercise are more likely to not be diagnosed with diabetes while those that do not exercise tend to be diagnosed with diabetes or prediabetes. From this right barplot, we can see that those who eat healthier are more likely to not be diagnosed with diabetes while those that do not eat healthy tend to be diagnosed with diabetes or prediabetes.

```
## [1] "Proportion for Healthy Eating and Exercise:"
```

```
##
##        0        1
## 0.581335 0.418665
```

```
## [1] "Proportions for Eating Poorly and No Exercise:"
```

```
##
##         0         1
## 0.3640227 0.6359773
```

Around 58.13% of those who eat healthy and exercise aren't diagnosed with diabetes while 41.86% are diagnosed with diabetes or prediabetes. Around 63.6% of those who don't eat healthy and don't exercise are diagnosed with diabetes while 36.40% are not diagnosed with diabetes.

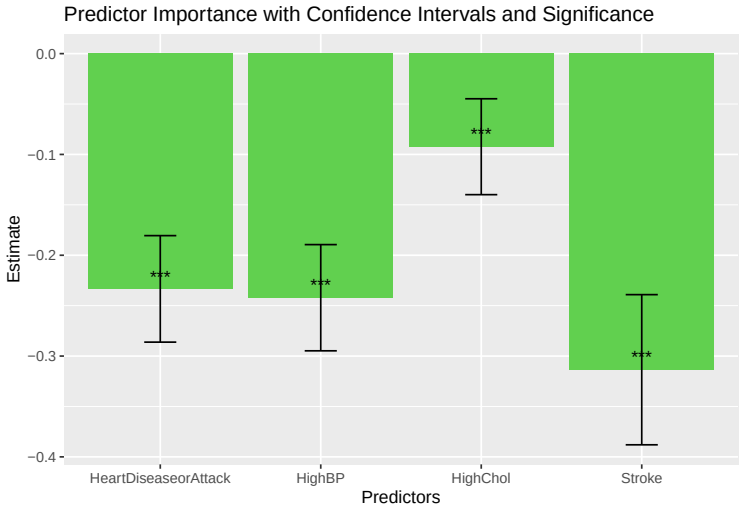
Conclusion

The combined evidence from logistic regression, t-tests, barplots, and proportion analysis strongly supports the conclusion that maintaining a healthy lifestyle, including regular physical activity and a diet rich in fruits and vegetables, is associated with a lower likelihood of diabetes. These findings emphasize the importance of promoting such lifestyle habits for diabetes prevention.

What do the intra-relationships look like within the physical factors such as an individuals blood pressure and cholesterol level?

We also want to see how physical health factors interact with each other. To do this, we look specifically at the subset of patients in our dataset who have diabetes.

We've found exercise to be a significant predictor of diabetes, along with fruit and vegetable consumption. If we look at how physical activity in those with diabetes can be influenced by other health conditions, we find the following:



High blood pressure, high cholesterol levels, past history of stroke, and past history of heart disease or heart attacks are all significant predictors of physical activity. More specifically, each variable has an inverse relationship with good physical activity, since the estimate of the intercept for each one is negative. That means that, for example, a person with high blood pressure is likely to have less physical activity, and a person with low blood pressure is likely to have more physical activity. And the same pattern occurs with other strong health conditions.

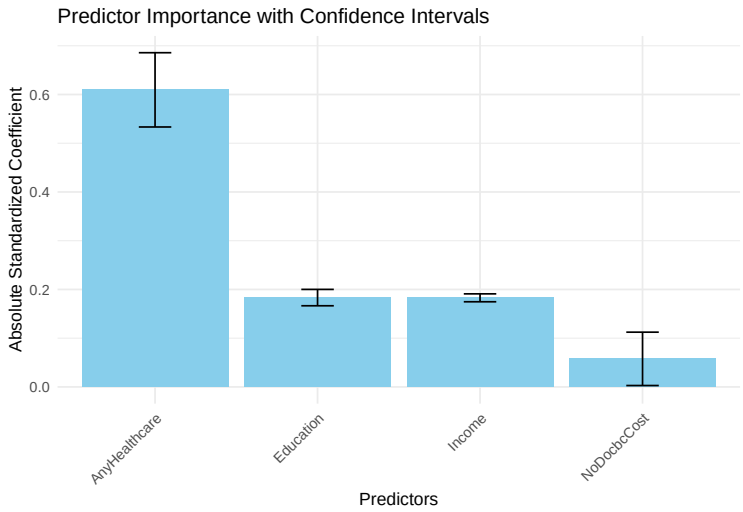
What socioeconomic factors have a significant impact on an individual's diagnosis of diabetes?

Socioeconomic factors from the dataset include the following:
Education: level of education from 1-6, with 1 being no education/kindergarten education to 6 being college education
Income: income level from 1-8 being different income brackets, 1 being the low-income bracket
AnyHealthcare: whether the respondent has any kind of health care coverage, such as health insurance and prepaid plans with 0 being no and 1 being yes

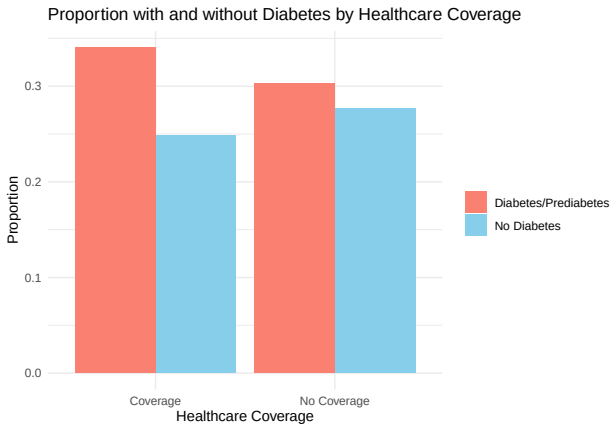
NoDocbcCost: whether the individual had a time within the past 12 months where they needed to see a doctor but couldn't because of the cost with 0 being no and 1 being yes

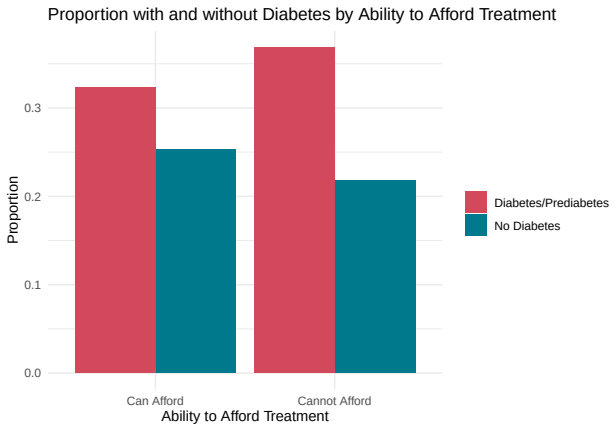
```
##
## Call:
## glm(formula = Diabetes_binary ~ Education + Income + NoDocbcCost +
##     AnyHealthcare, family = binomial, data = data)
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)    1.362468   0.052012  26.195   <2e-16 ***
## Education     -0.183284   0.008567 -21.393   <2e-16 ***
## Income        -0.182843   0.004148 -44.076   <2e-16 ***
## NoDocbcCost    0.057555   0.027928   2.061   0.0393 *
## AnyHealthcare  0.609471   0.038875  15.678   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 98000  on 70691  degrees of freedom
## Residual deviance: 93693  on 70687  degrees of freedom
## AIC: 93703
##
## Number of Fisher Scoring iterations: 4
```

Based on the p-values from the summary table of the generalized linear model of the predictor variable of interest, we can determine that Education, Income, AnyHealthcare and NoDocbcCost are all significant at a 0.05 level.



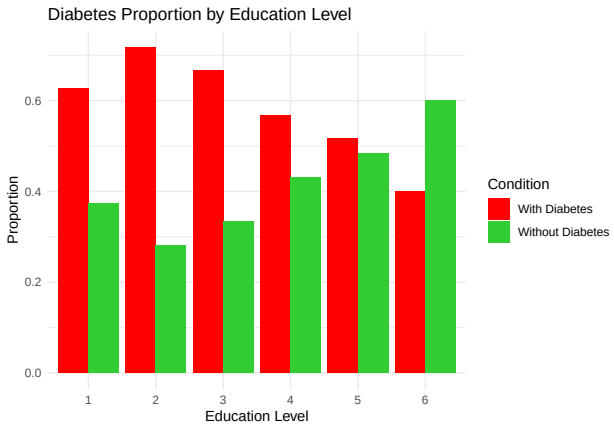
While all of the predictor variables are statistically significant, we can see here that AnyHealthcare (access to healthcare), is by far the strongest predictor for whether an individual is at risk/has diabetes. The error bars for the confidence interval provide a range of where the true value of the standardized coefficient will likely fall at a 95% confidence level. We will begin by looking at how access to healthcare influences the likelihood of an individual getting diabetes or not.

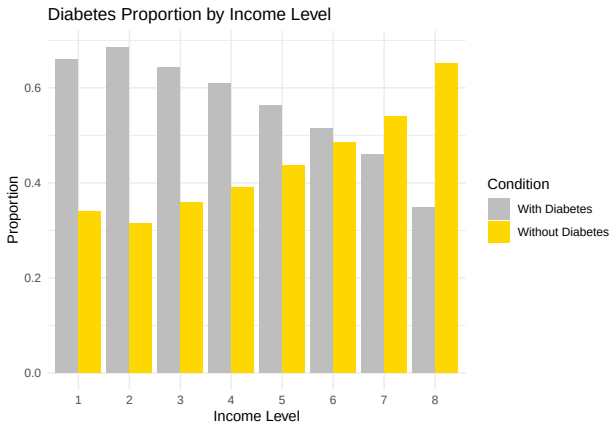




Here with the graph on the left we can see that there is actually a higher proportion of those with diabetes than those without in the category of individuals with healthcare coverage. This sounds un-intuitive, how can access to healthcare result in worse health? While the data is split 5050 for those reported with and without diabetes, the percent of individuals that have healthcare is reported to be over 95%. The dataset is very generous with its definition of healthcare coverage, described as "any kind of health care coverage", which can mean that the quality of the healthcare provided across individuals may differ drastically. In the United States where the data was collected, access to healthcare does not equate to medication and treatment procedures being covered by insurance and other coverage. We will next take a look at how being able to afford treatment can impact getting diabetes.

With the graph on the right we can see the proportion of those with and without diabetes by their ability to afford treatment from a doctor. The percentage of individuals that can actually afford treatment is slightly above 9%, a shockingly low amount considering how many had healthcare access. We can see that there is a significant difference in number of those with/without diabetes in the group of those that cannot afford treatment while the difference closes for those who can afford treatment.





The graph on the left shows the proportion of individuals diabetes by education level. The levels of education are the following: (1) Never attended school of only kindergarten (2) Grades 1-8 (3) Grades 9-11 (4) Grade 12 or GED (5) College 1 year to 3 years (or technical school) (6) College 4 years or more. We can see that there is a clear difference in those with and without diabetes through level of education. Individuals with an education between levels 1-3 are significantly more likely to have diabetes. Individuals between levels 4-5 close the gap significantly but still have more individuals with diabetes than without. At level 6, the image changes completely and there are far more individuals without diabetes. One of the major advantages that a higher education level can offer is a higher income.

The graph on the right shows the proportion of individuals diabetes by income level in thousands of USD. The levels of income are the following: (1) Less than 10 (2) 10-15 (3) 15-20 (4) 20-25 (5) 25-35 (6) 35-50 (7)

50-75 (8) Greater than 75. Like education, we can see that there is a significant difference of those with and without diabetes across income level. At income levels 7-8, the proportion of individuals without diabetes overtakes those with. A higher income level can result in the ability to afford more treatment options which can mean better health.

Conclusion

Utilizing the graphics and proportion analysis based on the logistic regression of these socioeconomic factors, we can conclude that having access to healthcare, the ability to afford treatment, education level, and income level drastically changes the likelyhood of an individual having diabetes. The section of individuals having access to healthcare was an interesting topic, as over 95% of individuals in the dataset had some access to healthcare but only 9% could afford treatment from a doctor. Splitting the respondant's healthcare options to tiers of increasing quality, similar to the levels of education and income, may provide more insightful information as to what quality of healthcare an individual needs before diabetes is significantly reduced.

What do the intra-relationships look like within the socioeconomic factors explored before?

The key predictor variable in socioeconomic factors is "AnyHealthcare," which indicates whether respondents have any form of healthcare. Consequently, we aim to investigate how access to healthcare influences other socioeconomic variables or how a person's economic status may affect their access to healthcare by examining additional factors.

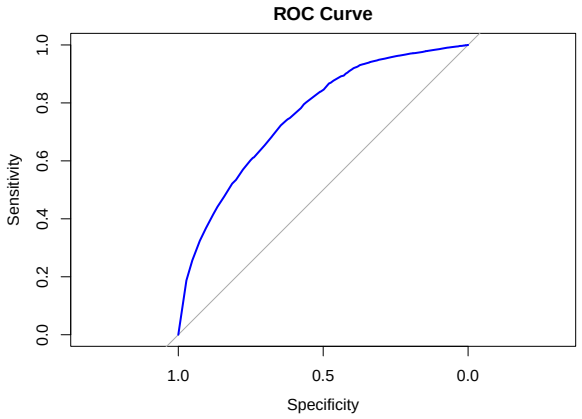
The economic factors we look at are income, education level, access to healthcare, and the ability to afford medical treatment.

```
##
## Call:
## glm(formula = AnyHealthcare ~ Income + Education + NoDocbcCost,
##     family = binomial, data = diabetes)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.476472   0.080011   18.45   <2e-16 ***
## Income       0.144889   0.009229   15.70   <2e-16 ***
## Education    0.255543   0.018424   13.87   <2e-16 ***
## NoDocbcCost -1.751819   0.040488  -43.27   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 25964  on 70691  degrees of freedom
## Residual deviance: 23043  on 70688  degrees of freedom
## AIC: 23051
##
## Number of Fisher Scoring iterations: 6
```

Income, education, and the ability to afford medical treatment are all significant predictors of access to healthcare under significance level $\alpha = 0.05$.

We can use the ROC curve to see the specificity and sensitivity of the model. We'd want a curve that fell close to the top left corner for the highest accuracy. The diagonal line in the ROC plot represents a random classifier. Points that lie above the diagonal line indicate the model having better-than-random performance.

We also look for an AUC (or area under the ROC curve) value greater than 0.5, since 0.5 also represents a random classifier.

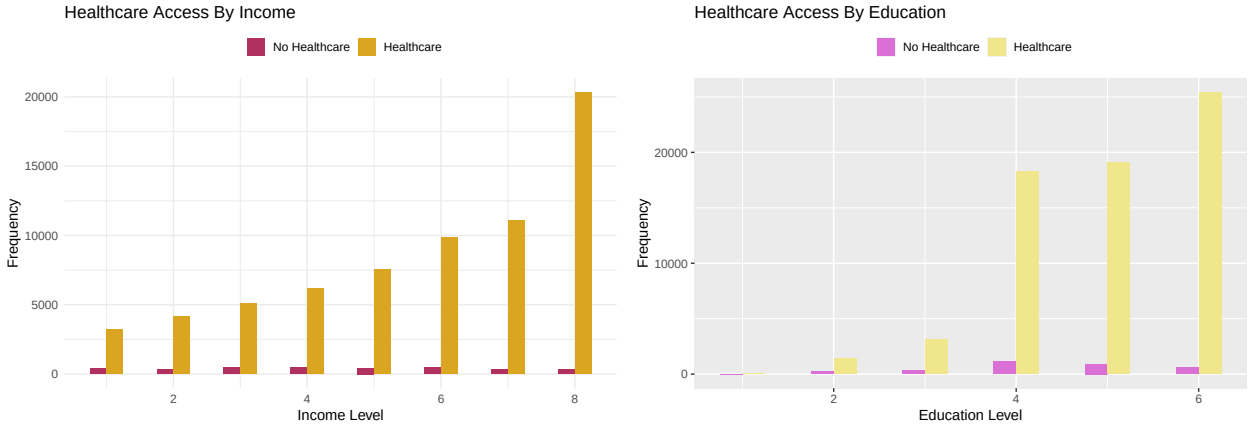



```
## Area under the curve: 0.757
```

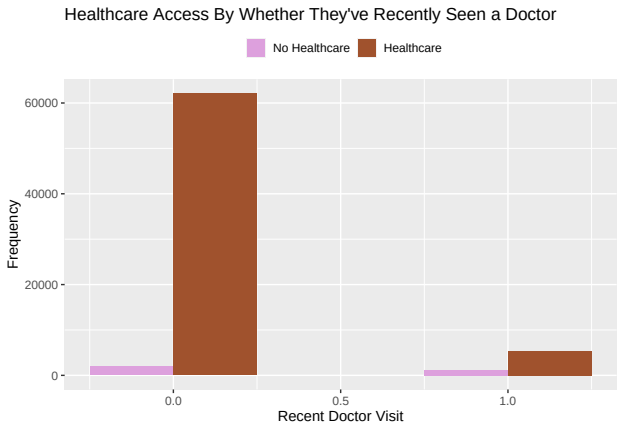
When looking at our ROC curve, the points are greater than the diagonal line, so we can say are model has good performance, but the ROC curve for our model does not fall close to the top left corner. This could mean that there is room for improvement in achieving a balance between sensitivity and specificity.

The AUC value is 0.757. We can interpret this to mean that the model is good at discriminating between classes. Since this value is greater than 0.5, we can say that the model is better than random chance.

If we want to look further into other inter-relationships in the data, we can also look at how healthcare interacts with the other economic predictors individually.



For both income and education, as the level increase which indicate higher levels of education or higher income brackets, we tend to see a higher frequency of people in that category having access to healthcare.



To analyze the graph barplot above, recent doctor visit signifies whether an individual had a time in the past 12 months where they needed to see a doctor but couldn't because of the financial cost where 0 indicates a "no" reponse and 1 indicates a "yes" response. With this barplot, we can see that people with access to healthcare were more likely to fall under the "no" response.

What's the best model for predicting whether an individual has diabetes considering collinearity effects?

Including an excessive number of predictor variables in a model is known to result in overfitting, which can inflate standard errors and render the model's results challenging to interpret. Additionally, the inclusion of all predictors may introduce new confounding paths, potentially leading to a less accurate model of the causal relationships we're hoping to analyze. With this question, we hope to test different models by taking out factors that may have collinearity with the k-fold CV method.

```
## Variable pair: GenHlth - Correlation: 0.5527567
## Variable pair: PhysHlth - Correlation: 0.4879758
```

A correlation value of around 0.5 or higher indicates a moderate to strong correlation. In our analysis, we've identified two pairs of factors around this range, demonstrating correlations of 0.488 and 0.553, respectively. The first pair involves "DiffWalk" and "PhysHlth," representing variables related to an individual's challenges in walking up stairs and the frequency of physical illnesses or injuries per month. The correlation between this pair has a correlation coefficient of 0.488. The second pair involves "GenHlth" and "PhysHlth," where "GenHlth" represents an individual's self-assessment of physical health on a scale from one to five, with one indicating excellent physical health. The correlation between "GenHlth" and "PhysHlth" is 0.553.

Since PhysHlth has a moderate correlation value with both GenHlth and DiffWalk, we will be testing the following models with k-fold cross validation:

Model 1: Diabetes prediction vs all predictors
Model 2: Diabetes prediction vs all predictors excluding PhysHlth
Model 3: Diabetes prediction vs all predictors excluding GenHlth
Model 4: Diabetes prediction vs all predictors excluding DiffWalk
Model 5: Diabetes prediction vs all predictors excluding GenHlth and DiffWalk
Model 6: Diabetes prediction vs all predictors excluding PhysHlth and DiffWalk
Model 7: Diabetes prediction vs all predictors excluding GenHlth and PhysHlth

Methodology

To accomplish this, the data set will be partitioned into six equally sized folds, ensuring that each fold contains an equivalent number of individuals with and without diabetes. Stratifying the folds by the diabetes

response enhances the effectiveness of cross-validation by maintaining a balanced representation of classes across folds. Following the creation of these stratified folds, a loop will iterate to train a model using five of the folds, with the mean squared error computed on the excluded fold. This process is repeated across all six folds, and the average MSE is calculated. The model yielding the lowest k-fold cross-validation estimate is considered the most effective. While leave-one-out cross validation could potentially provide more precise results due to its exhaustive nature, it becomes computationally impractical with a data set of over 70,000 observations due to its high computational cost.

Test Results

Model	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7
Cross_validation_error	0.2518955	0.2520794	0.2644797	0.2521304	0.2643326	0.2519861	0.2644769

Accuracy rate of model 1

Using model one, which was shown to have the least cross-validation error, a confusion matrix will be calculated to test the accuracy of our model.

```
##     Predicted
## True     0     1
##    0 25747  9599
##    1  8211 27135
```

Our accuracy can be calculated by first adding how many observations the model was able to accurately predict divided by the total number of observations in the dataset.

$$\text{Accuracy}=\frac{25747+27135}{70692} = \frac{52882}{70692} = 0.748$$

Our model one's accuracy is 74.8%, meaning that on average the model will accurately predict the diabetic status of an individual 74.8% of the time.

Conclusion

The outcomes from the six-fold cross-validation reveal that model one, where all predictors were utilized to predict diabetes, has the least cross-validation error. This superiority over the other models, which included excluded one or two predictors that had collinear effects, suggests that the standard error from these collinear predictors were not statistically significant. Despite introducing standard error into the logistic model, the benefit of having additional predictors seem to outweigh the drawbacks of the increased standard error. This was not the expected result as one would've expected removing predictors that were collinear would increase the model efficiency due to the removal of the standard error,

In our case, the multicollinearity seem to not be severe enough to cause significant issues where removing predictors would result in a more accurate model. Additionally, another explanation for this unexpected observation is that our sample size is sufficiently large enough to be able to predict the diabetic status of an individual even in the presence of collinearity. With a larger sample size, the estimates of the coefficients of the model tend to have lower standard errors and the increased degrees of freedom make the model less sensitive to collinearity.

Appendix

```
knitr::opts_chunk$set(echo = FALSE)
library(readr)
library(ggplot2)
library(dplyr)
library(broom)
library(pROC)
library(gridExtra)
library(tidyverse)
library(caret)
library(knitr)
library(tidyr)

setwd("/Users/justinlee/Desktop/")
data=read.csv("diabetes5050.csv")
diabetes=read.csv("diabetes5050.csv")
dat=read.csv("diabetes5050.csv")

log_reg=glm(Diabetes_binary~PhysActivity+Fruits+Veggies,family=binomial,
            data=data)
summary(log_reg)
# positively mostly correlated is veggies and fruits
# mostly negatively correlated us PhysActivity and Diabetes_binary


# Performing t-test to test the statistical significance of physical activity
# levels on the response variable diabetes_binary.
t.test(PhysActivity ~ Diabetes_binary, data = data)

# Performing t-test to test the statistical significance of vegetable
# consumption on the response variable diabetes_binary.
t.test(Veggies~ Diabetes_binary, data = data)

# Performing t-test to test the statistical significance of fruit consumption
# on the response variable diabetes_binary.
t.test(Fruits~ Diabetes_binary, data = data)

# Bar plot for exercise and diabetes diagnosis
barplot(table(data$Diabetes_binary, data$PhysActivity),
        beside = TRUE, col = c("lightblue", "lightgreen"),
        main = "Diabetes Diagnosis By Whether They Exercise",
        xlab = "Exercise (0:No, 1:Yes)", ylab = "Frequency",width = 0.5)
legend("topleft", legend = c("No Diabetes", "Diabetes or Prediabetes"),
       fill = c("lightblue","lightgreen"))
# Bar chart for healthy eating and diabetes diagnosis
combined_table <- table(data$Diabetes_binary, data$Veggies) +
  table(data$Diabetes_binary, data$Fruits)
# Barplot for healthy eating and diabetes diagnosis
barplot(
  combined_table,
  beside = TRUE,
  col = c("lightblue", "lightgreen"),
```

```
  main = "Diabetes Diagnosis by Eating Habits",
  xlab = "Healthy Eating Stats (0:No, 1:Yes)",
  ylab = "Frequency",
  ylim = c(0, 50000)
)
legend("topleft", legend = c("No Diabetes",
"Diabetes or Prediabetes"), fill = c("lightblue", "lightgreen"))

# finding table
healthy_exercise_lot <- subset(data, PhysActivity == 1 & Fruits == 1&Veggies==1)
poor_exercise_less <- subset(data, PhysActivity == 0 & Fruits == 0 & Veggies==0)

# for those who eat poorly and exercise less and who exercise more
prop_table_healthy <- prop.table(table(healthy_exercise_lot$Diabetes_binary))
prop_table_poor <- prop.table(table(poor_exercise_less$Diabetes_binary))

# printing proportions
print("Proportion for Healthy Eating and Exercise:")
print(prop_table_healthy)

print("Proportions for Eating Poorly and No Exercise:")
print(prop_table_poor)

#subsetting for responses where individuals had diabetes or prediabetes
diabetes_subset = diabetes %>% filter(Diabetes_binary == 1)

phys_conditions = glm(PhysActivity ~ HighBP+HighChol+Stroke+HeartDiseaseorAttack, data = diabetes_subse

# plotting confidence interval and significance for the predictors used in glm()
coefficients = coef(summary(phys_conditions))
results = data.frame(term = rownames(coefficients),
                     estimate = coefficients[, 1],
                     std.error = coefficients[, 2],
                     conf.low = coefficients[, 1] - 1.96 * coefficients[, 2],
                     conf.high = coefficients[, 1] + 1.96 * coefficients[, 2],
                     p.value = coefficients[, 4])
results = results[results$term != "(Intercept)", ]

ggplot(results, aes(x = term, y = estimate, ymin = conf.low, ymax = conf.high)) +
  geom_col(fill = 3) +
  geom_errorbar(width = 0.2) +
  geom_text(aes(label = ifelse(p.value < 0.05, "***", "")), vjust = -0.5) +
  labs(title = "Predictor Importance with Confidence Intervals and Significance",
       x = "Predictors",
       y = "Estimate")

# Creating a subset of the dataset consisting of the variables we will be
# working with
subset_data = data[, c("Diabetes_binary","Education","Income","NoDocbcCost",
                       "AnyHealthcare")]

glm_soc_econ = glm(formula = Diabetes_binary ~ Education + Income +NoDocbcCost+
                     AnyHealthcare, family = binomial, data = data)
```

```r
summary(glm_soc_econ)

tidy_glm <- broom::tidy(glm_soc_econ, conf.int = TRUE) %>%
  #Exclude Intercept
  filter(term != "(Intercept)") %>%
  # Calculate absolute values of the standardized coefficients
  mutate(estimate = abs(estimate),
         conf.low = abs(conf.low),
         conf.high = abs(conf.high))

# create a bar plot with error bars
ggplot(tidy_glm, aes(x = term, y = estimate)) +
  geom_bar(stat = "identity", position = position_dodge(), fill = "skyblue") +
  geom_errorbar(aes(ymin = conf.low, ymax = conf.high), width = 0.2,
                position = position_dodge(.9)) +
  labs(title = "Predictor Importance with Confidence Intervals",
       x = "Predictors",
       y = "Absolute Standardized Coefficient") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

# Calculate the proportions
healthcare_diabetes_table <- data %>%
  group_by(AnyHealthcare) %>%
  summarize(
    NoDiabetes = mean(Diabetes_binary == 0),
    Diabetes = mean(Diabetes_binary == 1)
  ) %>%
  mutate(
    NoDiabetes = NoDiabetes / sum(NoDiabetes + Diabetes),
    Diabetes = Diabetes / sum(NoDiabetes + Diabetes)
  ) %>%
  ungroup()

# Rename the factor levels
healthcare_diabetes_table$AnyHealthcare=
  recode(healthcare_diabetes_table$AnyHealthcare, `0` = 'No Coverage',
         `1` = 'Coverage')

# Format for ggplot
healthcare_diabetes_long <- healthcare_diabetes_table %>%
  pivot_longer(cols = c(NoDiabetes, Diabetes), names_to = "Condition",
               values_to = "Proportion")

# Create the bar graph
ggplot(healthcare_diabetes_long, aes(x = AnyHealthcare, y = Proportion,
                                     fill = Condition)) +
  geom_bar(stat = "identity", position = position_dodge()) +
  scale_fill_manual(values = c("NoDiabetes" = "skyblue", "Diabetes" = "salmon"),
                    labels = c("NoDiabetes" = "No Diabetes",
                               "Diabetes" = "Diabetes/Prediabetes")) +
  labs(title = "Proportion with and without Diabetes by Healthcare Coverage",
       x = "Healthcare Coverage",
```

```
       y = "Proportion") +
  theme_minimal() +
  theme(legend.title = element_blank())

# Calculate the proportions for NoDocbcCost
nodoc_summary <- data %>%
  group_by(NoDocbcCost) %>%
  summarize(
    NoDiabetes = mean(Diabetes_binary == 0),
    Diabetes = mean(Diabetes_binary == 1)
  ) %>%
  mutate(
    NoDiabetes = NoDiabetes / sum(NoDiabetes + Diabetes),
    Diabetes = Diabetes / sum(NoDiabetes + Diabetes)
  ) %>%
  ungroup()

# Rename the factor levels
nodoc_summary$NoDocbcCost=recode(nodoc_summary$NoDocbcCost,`0`='Can Afford',
                                 `1` = 'Cannot Afford')

# Format for ggplot
nodoc_long <- nodoc_summary %>%
  pivot_longer(cols = c(NoDiabetes, Diabetes), names_to = "Condition",
               values_to = "Proportion")

# Create the bar graph
ggplot(nodoc_long, aes(x = NoDocbcCost, y = Proportion, fill = Condition)) +
  geom_bar(stat = "identity", position = position_dodge()) +
  scale_fill_manual(values = c("NoDiabetes" = "#00798c",
                               "Diabetes" = "#d1495b"),
                    labels = c("NoDiabetes" = "No Diabetes",
                               "Diabetes" = "Diabetes/Prediabetes")) +
  labs(title="Proportion with and without Diabetes by Ability to Afford Treatment",
       x = "Ability to Afford Treatment",
       y = "Proportion") +
  theme_minimal() +
  theme(legend.title = element_blank())

# Summarize the data by Education level
education_summary <- data %>%
  group_by(Education) %>%
  summarize(
    Count = n(),
    Diabetes = sum(Diabetes_binary == 1),
    NoDiabetes = sum(Diabetes_binary == 0)
  ) %>%
  mutate(
    ProportionWithDiabetes = Diabetes / Count,
    ProportionWithoutDiabetes = NoDiabetes / Count
  ) %>%
  ungroup()
```

```r
# Transform the data from wide to long format
education_long <- education_summary %>%
  pivot_longer(cols = c(ProportionWithDiabetes, ProportionWithoutDiabetes),
               names_to = "Condition", values_to = "Proportion")

# Create the bar graph for Education
ggplot(education_long, aes(x = as.factor(Education), y = Proportion,
                           fill = Condition)) +
  geom_bar(stat = "identity", position = position_dodge()) +
  scale_fill_manual(values = c("ProportionWithDiabetes" = "red",
                               "ProportionWithoutDiabetes" = "limegreen"),
  labels = c("ProportionWithDiabetes" = "With Diabetes",
             "ProportionWithoutDiabetes" = "Without Diabetes")) +
  labs(title = "Diabetes Proportion by Education Level",
       x = "Education Level",
       y = "Proportion") +
  theme_minimal()

# Summarize the data by Income level
income_summary <- data %>%
  group_by(Income) %>%
  summarize(
    Count = n(),
    Diabetes = sum(Diabetes_binary == 1),
    NoDiabetes = sum(Diabetes_binary == 0)
  ) %>%
  mutate(
    ProportionWithDiabetes = Diabetes / Count,
    ProportionWithoutDiabetes = NoDiabetes / Count
  ) %>%
  ungroup()

# Transform the data from wide to long format
income_long <- income_summary %>%
  pivot_longer(cols = c(ProportionWithDiabetes, ProportionWithoutDiabetes),
               names_to = "Condition", values_to = "Proportion")

# Create the bar graph for Income
ggplot(income_long, aes(x = as.factor(Income), y = Proportion,
                        fill = Condition)) +
  geom_bar(stat = "identity", position = position_dodge()) +
  scale_fill_manual(values = c("ProportionWithDiabetes" = "gray",
                               "ProportionWithoutDiabetes" = "gold"),
                    labels = c("ProportionWithDiabetes" = "With Diabetes",
                               "ProportionWithoutDiabetes"="Without Diabetes"))+
  labs(title = "Diabetes Proportion by Income Level",
       x = "Income Level",
       y = "Proportion") +
  theme_minimal()

# general logistic regression with anyhealthcare as response variable and income
# education, nodocbccost being used as predictors
econ_log = glm(AnyHealthcare ~ Income + Education + NoDocbcCost,
```

```
               data = diabetes, family = binomial)
summary(econ_log)

# uses roc function to draw the roc curve
roc_curve = roc(diabetes$AnyHealthcare, econ_log$fitted.values)
plot(roc_curve, col = "blue", main = "ROC Curve", lwd = 2)

# uses the auc function to calculate the area under our roc curve
auc_value = auc(roc_curve); auc_value

# a bar plot of access to healthcare by income level
diabetes$AnyHealthcare <- factor(diabetes$AnyHealthcare, levels = c(0, 1),
                                 labels = c("No Healthcare", "Healthcare"))
ggplot(diabetes, aes(x = Income, fill = factor(AnyHealthcare))) +
  geom_bar(position = "dodge", width = 0.5) +
  labs(title = "Healthcare Access By Income",
       x = "Income Level",
       y = "Frequency") +
  scale_fill_manual(values = c("No Healthcare" = "maroon",
                               "Healthcare" = "goldenrod"),
                    name = "AnyHealthcare") +
  theme_minimal() +
  theme(legend.position = "top") +
  guides(fill = guide_legend(title = NULL, keywidth = 1, keyheight = 1))

# a bar plot of access to healthcare by education level
ggplot(diabetes, aes(x = Education, fill = factor(AnyHealthcare))) +
  geom_bar(position = "dodge", width = 0.5) +
  labs(title = "Healthcare Access By Education",
       x = "Education Level",
       y = "Frequency") +
  scale_fill_manual(values = c("No Healthcare"="orchid","Healthcare" = "khaki"),
                    name = "AnyHealthcare") +
  theme(legend.position = "top") +
  guides(fill = guide_legend(title = NULL, keywidth = 1, keyheight = 1))

# barplot of access to healthcare by the variable nodocbcost (variable explained
# in analysis)
ggplot(diabetes, aes(x = NoDocbcCost, fill = factor(AnyHealthcare))) +
  geom_bar(position = "dodge", width = 0.5) +
  labs(title = "Healthcare Access By Whether They've Recently Seen a Doctor",
       x = "Recent Doctor Visit",
       y = "Frequency") +
  scale_fill_manual(values = c("No Healthcare" = "plum",
                               "Healthcare" = "sienna"),
                    name = "AnyHealthcare") +
  theme(legend.position = "top") +
  guides(fill = guide_legend(title = NULL, keywidth = 1, keyheight = 1))

# Create correlation matrix with all variables except the first which happens
# to be the response variable
cor_matrix = cor(dat[,-1])
```

```r
# Set lower triangle and diagonal to NA to exclude duplicates and correlation
# coefficients of one
cor_matrix[lower.tri(cor_matrix, diag = TRUE)] = NA

# Find the top 2 correlation coefficients excluding duplicates and 1
top_cor_values = sort(unique(cor_matrix), decreasing = TRUE)[1:2]

# Find the indices of the top 2 correlation values
top_cor_indices = sapply(top_cor_values, function(value) {
  which(cor_matrix == value, arr.ind = TRUE)
})

for (i in seq_along(top_cor_values)) {
  index = top_cor_indices[, i]
  variable_pair = rownames(cor_matrix)[index[1]]
  correlation_value = cor_matrix[index[1], index[2]]

  # Display the top 2 correlation values
  cat("Variable pair:", variable_pair, "- Correlation:", correlation_value,"\n")
}

# Divide the dataset into six equal folds with an equal number of individuals
# with and without diabetes in each fold
k1 = dat[c(1:5891,35347:41237),]
k2 = dat[c(5892:11782,41238:47128),]
k3 = dat[c(11783:17673,47129:53019),]
k4 = dat[c(17674:23564,53020:58910),]
k5 = dat[c(23565:29455,58911:64801),]
k6 = dat[c(29456:35346,64802:70692),]

# 6-fold cross validation function
sixfold = function(model){
  n = nrow(dat)
  res = 0 # Initialization
  # Loops through 1 to 6, creating a train and test data set by rbinding the
  # folds that were created earlier
  for(k in 1:6){
    if(k==1){
      train = rbind(k2,k3,k4,k5,k6)
      test = k1
    }
    if(k==2){
      train = rbind(k1,k3,k4,k5,k6)
      test = k2
    }
    if(k==3){
      train = rbind(k1,k2,k4,k5,k6)
      test = k3
    }
    if(k==4){
      train = rbind(k1,k2,k3,k5,k6)
      test = k4
    }
```

```
  if(k==5){
    train = rbind(k1,k2,k3,k4,k6)
    test = k5
  }
  if(k==6){
    train = rbind(k1,k2,k3,k4,k5)
    test = k6
  }
  # Creates glm models that include the right predictors we want depending
  # on what is inputted as the model number
  if(model==1){
    glm.fit <- glm(Diabetes_binary ~ HighBP + HighChol + CholCheck + BMI +
                     Smoker + Stroke + HeartDiseaseorAttack + PhysActivity +
                     Fruits + Veggies + HvyAlcoholConsump + AnyHealthcare +
                     NoDocbcCost + MentHlth+Sex + Age + Education + Income +
                     PhysHlth+GenHlth + DiffWalk, data=train,family=binomial)
  }
  if(model==2){
    glm.fit <- glm(Diabetes_binary ~ HighBP + HighChol + CholCheck + BMI +
                     Smoker + Stroke + HeartDiseaseorAttack + PhysActivity +
                     Fruits + Veggies + HvyAlcoholConsump + AnyHealthcare +
                     NoDocbcCost + MentHlth + Sex + Age + Education + Income +
                     GenHlth + DiffWalk, data=train, family=binomial)
  }
  if(model==3){
    glm.fit <- glm(Diabetes_binary ~ HighBP + HighChol + CholCheck + BMI +
                     Smoker + Stroke + HeartDiseaseorAttack + PhysActivity +
                     Fruits + Veggies + HvyAlcoholConsump + AnyHealthcare +
                     NoDocbcCost + MentHlth + Sex + Age + Education + Income +
                     DiffWalk + PhysHlth, data=train, family=binomial)
  }
  if(model==4){
    glm.fit <- glm(Diabetes_binary ~ HighBP + HighChol + CholCheck + BMI +
                     Smoker + Stroke + HeartDiseaseorAttack + PhysActivity +
                     Fruits + Veggies + HvyAlcoholConsump + AnyHealthcare +
                     NoDocbcCost + MentHlth + Sex + Age + Education + Income+
                     PhysHlth + GenHlth, data=train, family=binomial)
  }
  if(model==5){
    glm.fit <- glm(Diabetes_binary ~ HighBP + HighChol + CholCheck + BMI +
                     Smoker + Stroke + HeartDiseaseorAttack + PhysActivity +
                     Fruits + Veggies + HvyAlcoholConsump + AnyHealthcare +
                     NoDocbcCost + MentHlth + Sex + Age + Education + Income+
                     PhysHlth, data=train, family=binomial)
  }
  if(model==6){
    glm.fit <- glm(Diabetes_binary ~ HighBP + HighChol + CholCheck + BMI +
                     Smoker + Stroke + HeartDiseaseorAttack + PhysActivity +
                     Fruits + Veggies + HvyAlcoholConsump + AnyHealthcare +
                     NoDocbcCost + MentHlth + Sex + Age + Education + Income+
                     GenHlth, data=train, family=binomial)
  }
  if(model==7){
```

```
      glm.fit <- glm(Diabetes_binary ~ HighBP + HighChol + CholCheck + BMI +
                       Smoker + Stroke + HeartDiseaseorAttack + PhysActivity +
                       Fruits + Veggies + HvyAlcoholConsump + AnyHealthcare +
                       NoDocbcCost +MentHlth + Sex + Age + Education + Income+
                       DiffWalk, data=train, family=binomial)
    }
    # Uses the model that is inputted to predict the diabetes variable using the
    # test data set that was created earlier. Classifies the diabetes response
    # as 1 if the y_pred is greater than 0.5 and 0 otherwise.
    y_pred = predict(glm.fit, data = test, type = "response")
    y_res = ifelse(y_pred > 0.5, 1, 0)
    # Calculating the model's MSE
    res = res + mean((y_res - test$Diabetes_binary)^2)
  }
  # Averaging the MSE across the fold iterations.
  res <- res/6
  return(res)
}

# Store the cross validation errors
mod1 = sixfold(model = 1)
mod2 = sixfold(model = 2)
mod3 = sixfold(model = 3)
mod4 = sixfold(model = 4)
mod5 = sixfold(model = 5)
mod6 = sixfold(model = 6)
mod7 = sixfold(model = 7)

# Creating a data frame to store the results
results_df <- data.frame(
  Model = c("Model 1", "Model 2", "Model 3", "Model 4", "Model 5",
            "Model 6", "Model 7"),
  Cross_validation_error = c(mod1, mod2, mod3, mod4, mod5, mod6, mod7)
)

# Create an easy to read table with the stored cross validation errors and
# which model they belong to
as.data.frame(t(results_df)) %>% kable(col.names=NULL)

# Recreates model 1 that uses all predictors
model1 = glm(Diabetes_binary ~ HighBP + HighChol + CholCheck + BMI + Smoker +
               Stroke + HeartDiseaseorAttack + PhysActivity + Fruits + Veggies+
               HvyAlcoholConsump + AnyHealthcare + NoDocbcCost + MentHlth + Sex+
               Age + Education + Income + DiffWalk + GenHlth + PhysHlth,
             data=dat, family=binomial)
table(
  dat$Diabetes_binary,
  # Uses model 1 to predict the diabetes response variable of the original
  # dataset and classifies it as "1" if it predicts it to be >0.5
  ifelse(predict(model1, data = dat, type="response") > 0.5, 1, 0),
  dnn = c("True", "Predicted")
)
```